Optimization of multi-fidelity computer experiments via the EQIE criterion

Xu He, Rui Tuo
Academy of Mathematics and Systems Science,
Chinese Academy of Sciences
and C. F. Jeff Wu
School of Industrial and Systems Engineering,
Georgia Institute of Technology

January 4, 2016

Abstract

Computer experiments based on mathematical models are powerful tools for understanding physical processes. This article addresses the problem of kriging-based optimization for deterministic computer experiments with tunable accuracy. Our approach is to use multi-fidelity computer experiments with increasing accuracy levels and a nonstationary Gaussian process model. We propose an optimization scheme that sequentially adds new computer runs by following two criteria. The first criterion, called EQI, scores candidate inputs with given level of accuracy, and the second criterion, called EQIE, scores candidate combinations of inputs and accuracy. From simulation results and a real example using finite element analysis, our method outperforms the expected improvement (EI) criterion which works for single-accuracy experiments.

Keywords: Expected improvement criterion, Gaussian process model, Kriging, design of experiment, multi-fidelity experiment

1 INTRODUCTION

Computer simulation based on mathematical models such as partial differential equations is a powerful tool for understanding physical processes. The computer code being used is often referred to as a *simulator*. Although running a computer code is more affordable than performing a physical experiment, it is usually infeasible to obtain outputs for all possible or most combinations of the inputs of a simulator. The area of *computer experiments*, as an emerging branch of statistics, is primarily concerned with the development of methodologies for the design and analysis of computer simulations. In computer experiments, surrogate models, also referred to as *emulators*, are usually employed to predict the outputs of the simulators at untried input points. Gaussian process models are common choices for these emulators (Sacks et al., 1989; Santner et al., 2003).

One major problem in computer experiments is to find the inputs which produce the most desirable outcome. An optimization procedure usually employs two major steps. First, some initial outputs are obtained from the simulator, with inputs commonly chosen using space-filling designs (Mckay et al., 1979; Tang, 1993). Second, the emulator is updated iteratively to search for the best inputs. The expected improvement (EI) criterion (Jones et al., 1998) is widely used to score candidate inputs. Given a new set of inputs $\mathbf{x}$, EI($\mathbf{x}$) gives the expected improvement at $\mathbf{x}$ on the output from the best achieved inputs.

Computer experiments usually return approximate solutions of the physical model. For example, in finite element analysis, the physical model is usually given by partial differential equations (Brenner and Scott, 2007). Because the equations are solved by numerical algorithms, it cannot reach the exact solution of the physical models in finite time. The error between the exact solution and the approximate solution is referred to as "discretization error" in Chkrebtii et al. (2014). Usually, the accuracy of simulators is controlled by tuning parameters. In finite element analysis, one key parameter is the mesh size h. For simplicity, throughout this work we assume there is only one tuning parameter. Higher accuracy simulations produce less error, but require more computing resources.

However, the EI criterion is limited to single-accuracy computer experiments. To employ the EI criterion, we need to fix the value of h and treat the outputs from the simulator as the exact solution. For computer experiments with tunable accuracy, this approach is not

suitable. First, as demonstrated from the simulation results in Section 4, performance of optimization algorithms is sensitive to the choice of tuning parameter. It is not desirable to choose a suboptimal tuning parameter value. By employing multi-fidelity simulation runs, we can quantify the magnitude of errors and costs and choose h adaptively in the process of optimization. We can also obtain a measure of uncertainty on the final result. Second, as we gain more knowledge on the simulator, we need fewer but more accurate simulations. Therefore the initial simulations should be assigned to cheaper runs but with lower accuracy, the global search simulations should be assigned to those with medium accuracy and the local refinement simulations should be assigned to those with higher accuracy. Thus, it is advantageous to run computer experiments with increasing accuracies.

As an extension of the EI criterion, Picheny et al. (2013) proposed the *expected quantile improvement* (EQI) criterion for stochastic computer experiments with different accuracy levels. Given a prespecified h and a new set of inputs $\mathbf{x}$, EQI($\mathbf{x}$) gives the expected improvement at $\mathbf{x}$ on the output, up to a confidence level, from the best achieved inputs. In their work, the errors of the computer experiments are assumed to be mutually independent. An adaption of the EQI criterion for partially converged simulations using the model by Picheny and Ginsbourger (2013) is discussed briefly in the rejoinder of Picheny et al. (2013). There are several other works which employ multi-fidelity computer experiments. Huang et al. (2006) and Forrester et al. (2007) consider Gaussian process models in which h is a qualitative variable. In their work, the highest-fidelity outputs are treated as the exact solution and outputs from each fidelity are modeled separately. As a result, the choice of h for future runs are confined to the h for initial computer runs and their methods are mostly applied to experiments with two levels of accuracy. Our work is based on the nonstationary Gaussian process model proposed by Tuo, Wu and Yu (2014) in which h is a numerical variable. This nonstationary model is more suitable for experiments with more than two levels of fidelity. Furthermore, this model enables us to quantify the accuracy of the highest-fidelity runs. As a result, we can keep increasing the accuracy of runs. Based on this model, we propose our EQI criterion for scoring candidate inputs. Because of the different assumptions on the error terms, our EQI criterion has different formulas and properties from the original EQI criterion.

Although the EQI criterion is effective for scoring candidate inputs, none of the above methods can recommend the accuracy level of candidate inputs. In this paper, we propose the *expected quantile improvement efficiency* (EQIE) criterion that scores the candidate combination of $(\mathbf{x}, h)$. Its intent is to optimize the expected quantile improvement on the unobserved exact solution. Our EQIE criterion consists of two parts, the EQI part and the cost function. By incorporating a cost function, the EQIE criterion can strike a proper balance between computational accuracy and cost. Similar to the EI criterion, it also balances between global search and local refinement. Furthermore, the EQIE criterion tends to recommend higher accuracy levels for local refinement runs than for global search runs. Because of these advantages, algorithms using EQIE provide better results on average than algorithms using EI or EQI in our simulation studies and a real example, with the same budget on total costs.

We review the EQI criterion from Picheny et al. (2013) and the nonstationary model from Tuo et al. (2014) in Section 2. We introduce our EQI and EQIE criteria in Section 3. In Section 4, we evaluate the performance of the proposed criteria by simulation studies. A real example in casting simulations is given in Section 5 to illustrate and compare the proposed approach with existing methods. Conclusions and further remarks are given in Section 6. Derivations are given in the supplementary materials.

2 REVIEW OF SOME RELEVANT RESULTS

In this section, we review the EQI criterion from Picheny et al. (2013) and the nonstationary Gaussian process model from Tuo et al. (2014).

2.1 The EQI criterion by Picheny et al.

Picheny et al. (2013) proposed the EQI criterion for stochastic simulations. Denote the quantity of interest at input point x as $f(x)$. In a stochastic simulation, the quantity of interest cannot be obtained in a finite time. Instead, a random approximation of $f(x)$ can be given by the computer code, usually through Monte Carlo simulation. The accuracy of the random output can be controlled by the number of iterations of the simulation. Usually,

this accuracy can be quantified by a real number, denoted by h, which is determined by the number of iterations or the computational time. As the number of iterations or the computational time goes to infinity, h tends to zero. Given x and h, the output of the computer code can then be modeled as

$$y(x, h) = f(x) + e(h), \tag{1}$$

where $e(h) \sim N(0, h)$. Thus, $y(x, h) \to f(x)$ as $h \to 0$. Picheny et al. (2013) also models $f(x)$ as a realization of a stationary Gaussian process model, which allows one to predict $f(x)$ given a set of computer outputs $\{y(x_1, h_1), \ldots, y(x_n, h_n)\}$.

A common scenario in practice is to find the minimizer/maximizer of f. Picheny et al. (2013) suggests a sequential approach, by choosing the next design point to be the one minimizing the EQI criterion.

$$\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1}) = \mathrm{E}\left(\{\min Q_n(\mathbf{x_i}) - Q_{n+1}(\mathbf{x_{n+1}})\}^+ | \mathbf{A}_n\right), \tag{2}$$

where $z^+ = \max\{z, 0\}$, $\mathbf{A}_n$ denote the collection of $\{\mathbf{x_1}, \ldots, \mathbf{x_n}, h_1, \ldots, h_n, y_1, \ldots, y_n\}$, $Q_n(\mathbf{x})$ is the αth quantile estimate of the exact solution based on $\mathbf{A}_n$, and $Q_{n+1}(\mathbf{x_{n+1}})$ is the αth quantile estimate of the exact solution based on $\mathbf{A}_n$ and one new trial $(\mathbf{x_{n+1}}, h_{n+1}, \zeta(\mathbf{x_{n+1}}, h_{n+1}))$. Explicit expressions for $\mathrm{EQI}(\mathbf{x}, h)$ can be found in their paper.

In this article, we consider the sequential design problem of deterministic computer experiments, usually in terms of finite element analysis. In this context the EQI method by Picheny et al. (2013) is not applicable because the model in (1) is not suitable for deterministic computer experiments.

2.2 A nonstationary Gaussian process model by Tuo et al.

Tuo et al. (2014) proposed the following model for computer experiments with multiple levels of accuracy. Let $\zeta(\mathbf{x}, h)$ be the computer output with inputs $\mathbf{x}$ and the tuning parameter h, where $\mathbf{x} \in \mathbf{R}^d$ and $h > 0$; $\varphi(\mathbf{x})$ be the exact solution to the physical model, and $\delta(\mathbf{x}, h)$ be the numerical error. These functions are linked as follows:

$$\zeta(\mathbf{x}, h) = \varphi(\mathbf{x}) + \delta(\mathbf{x}, h). \tag{3}$$

A standard modeling technique in computer experiments is to regard unknown functions as realizations of Gaussian processes. Specifically, they assume that $\varphi(\cdot)$ and $\delta(\cdot,\cdot)$ are realizations of independent Gaussian processes, denoted as $V(\cdot)$ and $Z(\cdot,\cdot)$. The means of the Gaussian processes are modeled as linear combinations of unknown functions, namely:

$$\mathrm{E}(V(\mathbf{x})) = \mathbf{f}_1^{\mathrm{T}}(\mathbf{x})\boldsymbol{\beta}_1, \quad \mathrm{E}(Z(\mathbf{x},h)) = \mathbf{f}_2^{\mathrm{T}}(h)\boldsymbol{\beta}_2, \tag{4}$$

where $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are unknown vectors of coefficients and $\mathbf{f}_2(0) = 0$. Suppose $V(\cdot)$ and $Z(\cdot,\cdot)$ have the covariance functions

$$\begin{aligned} \mathrm{Cov}(V(\mathbf{x_1}), V(\mathbf{x_2})) &= \sigma^2 K_{\boldsymbol{\theta_1}}(\mathbf{x_1}, \mathbf{x_2}), \\ \mathrm{Cov}(Z(\mathbf{x_1}, h_1), Z(\mathbf{x_2}, h_2)) &= \tau^2 K_{\boldsymbol{\theta_2}}(\mathbf{x_1}, \mathbf{x_2}) \min(h_1, h_2)^w, \end{aligned} \tag{5}$$

where $K_{\boldsymbol{\theta}}$ is a family of correlation functions indexed by $\boldsymbol{\theta}$ and w is a positive constant. Computer runs with higher accuracy are associated with lower h. The ratio τ^2/σ^2 gives the magnitude of numerical error relative to the variation of the exact solution (due to the variation of the input variables). One common choice is the Gaussian correlation family, with

$$K_{\boldsymbol{\theta}}(\mathbf{x_1}, \mathbf{x_2}) = \exp\left\{ -\sum_{i=1}^{d} \theta_i (x_{1,i} - x_{2,i})^2 \right\}, \tag{6}$$

where $\theta_i, x_{1,i}$ and $x_{2,i}$ denote the ith entry of $\boldsymbol{\theta}, \mathbf{x_1}$ and $\mathbf{x_2}$, respectively. For brevity, we shall refer to this model as the TWY model hereinafter.

From (5), it can be seen that Z is a nonstationary Gaussian process because its covariance function tends to 0 as h goes to 0. In this model, $\mathrm{Var}(Z(\mathbf{x},h))$ is monotonically increasing in h. We believe that this variance structure captures the essence of the multi-fidelity computer experiments because computer simulation over coarser mesh is likely to produce less accurate output, and the associated increase in uncertainty is modeled as an increase in the variance. Also equation (5) leads to a reduction in correlation between simulations at $\mathbf{x_1}$ and $\mathbf{x_2}$ as the difference between h_1 and h_2 increases. That is, we learn less about future high fidelity simulations from simulations at lower fidelity. Alternative choices of models to (5) and (6) can be found in Tuo et al. (2014). The main idea of this work can be similarly applied to these models. Our numerical experience shows that it does not render a much different result by adopting a different model. Therefore, we will focus on the model given by (5) and (6) from now on.

The value of w in (5) determines the rate at which $Z(\mathbf{x}, h)$ converges to 0 as h goes to 0. Tuo et al. (2014) suggests determining w by comparing this rate with the error bound of the corresponding numerical algorithm. For instance, $w = 2$ is appropriate for finite element methods (Tuo et al., 2014, Section 5).

Given a set of design points $\{(\mathbf{x_i}, h_i)\}_{i=1}^n$, let $y_i = \zeta(\mathbf{x_i}, h_i)$ be the computer response for each i. Tuo et al. (2014) discusses a Bayesian approach to predict for the exact solution $\varphi(\mathbf{x})$ at an untried input $\mathbf{x}$.

3 PROPOSED METHOD

In this section, we propose our EQI and EQIE criteria and algorithms that employ these criteria.

3.1 The EQI criterion for the TWY model

Our EQI criterion for the TWY model has the same definition as in (2):

$$\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1}) = \mathrm{E}\left(\{\min Q_n(\mathbf{x_i}) - Q_{n+1}(\mathbf{x_{n+1}})\}^+ | \mathbf{A}_n\right). \tag{7}$$

Because of the difference in the underlying model, our EQI criterion has different derivations and properties. We assume that $w > 0$, $\sigma^2 > 0$, $\tau^2 \geq 0$, $\boldsymbol{\theta}_1 > 0$ and $\boldsymbol{\theta}_2 > 0$. We discuss below the quantities $\min Q_n(\mathbf{x_i})$ and $Q_{n+1}(\mathbf{x_{n+1}})$ separately.

First, $\min Q_n(\mathbf{x_i})$ represents the expected optimal outcome of the exact solution, up to n trials. Let $l_n(\mathbf{x})$ and $r_n(\mathbf{x})$ denote the estimated mean of $\varphi(\mathbf{x})$ in (3) and the standard error of $l_n(\mathbf{x})$ based on $\mathbf{A}_n$, respectively. Then

$$Q_n(\mathbf{x}) = l_n(\mathbf{x}) + \Phi^{-1}(\alpha) r_n(\mathbf{x}), \tag{8}$$

where $\Phi(x)$ is the the cumulative distribution function of the standard normal distribution and $\alpha \geq 0.5$ is a fixed quantile level. Since the computer runs are carried out with different h, a reasonable choice of $\mathbf{x}$ is the $\mathbf{x_i}$ with minimal $Q_n(\mathbf{x_i})$. With $\mathbf{A}_n$ given, $Q_n(\mathbf{x_i})$ and $\min Q_n(\mathbf{x_i})$ are fixed quantities.

Next, assume that we carry out another trial with input $\mathbf{x_{n+1}}$, tuning parameter h_{n+1} and outcome $y_{n+1} = \zeta(\mathbf{x_{n+1}}, h_{n+1})$. Then $Q_{n+1}(\mathbf{x_{n+1}})$ is the expected outcome of the exact

solution for $\mathbf{x_{n+1}}$, up to $n+1$ trials. Similar to (8),

$$Q_{n+1}(\mathbf{x_{n+1}}) = l_{n+1}(\mathbf{x_{n+1}}) + \Phi^{-1}(\alpha) r_{n+1}(\mathbf{x_{n+1}}), \tag{9}$$

where $l_{n+1}(\mathbf{x})$ and $r_{n+1}(\mathbf{x})$ are the estimated mean of $\varphi(\mathbf{x})$ and standard error of $l_{n+1}(\mathbf{x})$ based on $\mathbf{A}_n \cup \{\mathbf{x_{n+1}}, h_{n+1}, y_{n+1}\}$, respectively. With $\mathbf{A}_n$, $\mathbf{x_{n+1}}$ and h_{n+1} given, $Q_{n+1}(\mathbf{x_{n+1}})$ is a random quantity because y_{n+1} is random. For an observed y_{n+1} value, if $Q_{n+1}(\mathbf{x_{n+1}}) < \min Q_n(\mathbf{x_i})$, then $\min Q_n(\mathbf{x_i}) - Q_{n+1}(\mathbf{x_{n+1}})$ is the improvement on the expected optimal outcome. Otherwise, there is no improvement.

For $(\mathbf{x_{n+1}}, h_{n+1}) \notin \{(\mathbf{x_i}, h_i) | i = 1, \ldots, n\}$, the EQI value can be expressed in a closed form as follows:

$$\begin{aligned}
& \mathrm{EQI}(\mathbf{x}_{n+1}, h_{n+1}) \\
= \; & \left[\min Q_n(\mathbf{x_i}) - \{\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}})) + \Phi^{-1}(\alpha) r_{n+1}(\mathbf{x_{n+1}})\}\right] \\
& \Phi\left(\frac{\min Q_n(\mathbf{x_i}) - \{\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}})) + \Phi^{-1}(\alpha) r_{n+1}(\mathbf{x_{n+1}})\}}{\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2}}\right) \\
& + \mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2} \phi\left(\frac{\min Q_n(\mathbf{x_i}) - \{\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}})) + \Phi^{-1}(\alpha) r_{n+1}(\mathbf{x_{n+1}})\}}{\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2}}\right)
\end{aligned} \tag{10}$$

where $\phi(x)$ is the probability density function of the standard normal distribution. Derivations and formulas for $l_n(\mathbf{x})$, $r_n(\mathbf{x})$, $l_{n+1}(\mathbf{x})$ and $r_{n+1}(\mathbf{x})$ are given in the supplementary materials. Because $r_{n+1}(\mathbf{x_{n+1}})$ has a closed form and $l_{n+1}(\mathbf{x_{n+1}})$ follows a normal distribution with mean and variance given in closed forms, $\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1})$ has a closed form.

When $\alpha = 0.5$, $\Phi^{-1}(\alpha) = 0$ and

$$\begin{aligned}
\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1}) \;=\; & \{\min Q_n(\mathbf{x_i}) - \mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))\} \, \Phi\left(\frac{\min Q_n(\mathbf{x_i}) - \mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))}{\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2}}\right) \\
& + \mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2} \phi\left(\frac{\min Q_n(\mathbf{x_i}) - \mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))}{\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2}}\right).
\end{aligned} \tag{11}$$

If there is no numerical error, i.e., $\delta(\mathbf{x_{n+1}}, h_{n+1}) = 0$, then $l_n(\mathbf{x_i}) = y_i$ and $r_n(\mathbf{x_i}) = 0$ for $i = 1, \ldots, n$, and $l_{n+1}(\mathbf{x_{n+1}}) = \zeta(\mathbf{x_{n+1}}, h_{n+1})$. Thus, $\min Q_n(\mathbf{x_i}) = \min(l_n(\mathbf{x_i})) = \min y_i$, $\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}})) = l_n(\mathbf{x_{n+1}})$, $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}})) = r_n(\mathbf{x_{n+1}})^2$ and

$$\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1}) = \{\min(y_i) - l_n(\mathbf{x_{n+1}})\} \Phi\left(\frac{\min(y_i) - l_n(\mathbf{x_{n+1}})}{r_n(\mathbf{x_{n+1}})}\right) + r_n(\mathbf{x_{n+1}}) \phi\left(\frac{\min(y_i) - l_n(\mathbf{x_{n+1}})}{r_n(\mathbf{x_{n+1}})}\right),$$

which is the EI criterion by Jones et al. (1998).

From (11), we can see that EQI with $\alpha = 0.5$ has two major parts, $\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))$ and $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))$. The EQI prefers inputs with low $\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))$ and high $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))$. Suppose $(\mathbf{x_1}, h_1, y_1)$ is a local minimum point. Then in a neighborhood of $\mathbf{x_1}$, both $\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))$ and $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))$ increases as $|\mathbf{x_{n+1}} - \mathbf{x_1}|$ increases. Inputs too close to $\mathbf{x_1}$ have low $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))$, while inputs too far away from $\mathbf{x_1}$ have high $\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))$. Neither is preferred by the EQI criterion. Furthermore, $\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1}) = 0$ if $(\mathbf{x_{n+1}}, h_{n+1}) = (\mathbf{x_1}, h_1)$. As a result, the local optimum of $\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1})$ is achieved in reasonably close distance from $\mathbf{x_1}$, and EQI with $\alpha = 0.5$ can balance between global exploration (i.e., $\mathbf{x_{n+1}}$ close to at least one $\mathbf{x_i}$) and local refinement (i.e., $\mathbf{x_{n+1}}$ far away from any $\mathbf{x_i}$).

With $\alpha \neq 0.5$, EQI has one more part, i.e., $r_{n+1}(\mathbf{x_{n+1}})$. The $r_{n+1}(\mathbf{x_{n+1}})^2$ term is the variance of $l_{n+1}(\mathbf{x_{n+1}}) - \varphi(\mathbf{x_{n+1}})$ after the new output $\zeta(\mathbf{x_{n+1}}, h_{n+1})$ is obtained. In a neighborhood of $\mathbf{x_1}$, $r_{n+1}(\mathbf{x_{n+1}})$ decreases as $|\mathbf{x_{n+1}} - \mathbf{x_1}|$ decreases. Therefore, for a higher α, our EQI criterion generally recommends new runs closer to the evaluated inputs. On the other hand, for a lower α, new runs recommended by the EQI criterion are generally exploratory. In our experience, EQI performs well with $0.5 \leq \alpha \leq 0.9$.

The $r_{n+1}(\mathbf{x_{n+1}})$ terms decrease as h_{n+1} decreases, the $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))$ term increases slightly as h_{n+1} decreases and the $\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}}))$ term is insensitive to h_{n+1}. Therefore, for $\alpha \geq 0.5$, EQI usually favors the lowest possible h_{n+1} value. As a result, the EQI criterion cannot be used to select the optimal h_{n+1}. In the studies in Sections 4 and 5, we set the value of h_{n+1} and use the EQI criterion with $\alpha = 0.5$ to find the best input.

3.2 The EQIE Criterion for the TWY model

In this subsection, we propose the Expected Quantile Improvement Efficiency (EQIE) criterion for scoring candidate inputs and the tuning parameter. The EQIE criterion is the expected improvement on the estimated quantile of the exact solution divided by the cost from obtaining the new outcome at $(\mathbf{x_{n+1}}, h_{n+1})$, that is,

$$\mathrm{EQIE}(\mathbf{x_{n+1}}, h_{n+1}) = \mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1}) / C(\mathbf{x_{n+1}}, h_{n+1}), \tag{12}$$

where $\mathrm{EQI}(\mathbf{x_{n+1}}, h_{n+1})$ is given in Section 3.1 and $C(\mathbf{x_{n+1}}, h_{n+1})$ is the cost of running the computer code at $(\mathbf{x_{n+1}}, h_{n+1})$. Note that a similar solution, called "augmented expected improvement", was proposed in Huang et al. (2006).

Usually, the cost depends on h_{n+1} but is less sensitive to $\mathbf{x_{n+1}}$. Thus we can use a cost function $C(h)$ depending only on h to approximate $C(\mathbf{x}, h)$. It will be shown that it suffices to know the order of magnitude of $C(h)$ with respect to h. In some situations, the time complexity of the numerical algorithm is known. For example, the computational time of an optimal finite element solver for an elliptic partial differential equation grows linearly with respect to h^{-r}, where r is the spatial dimension of the partial differential equation (Brenner and Scott, 2007). In other situations, the time complexity of the computer code is not known in advance. A common scenario is when the computer code is given by a commercial software, in which the computer model is a blackbox and the time complexity of the algorithm is also unknown. In this situation, we need to run the computer code at different h values to get an approximate estimate for the cost function. For example, in Example 3 of Tuo et al. (2014), it is seen that $C(h)$ is approximately proportional to h^{-4}. A constant term may be added to $C(h)$ to represent a fixed cost in running the simulation.

From (12), for fixed h_{n+1}, the EQIE criterion is equivalent to the EQI criterion. Therefore, the EQIE criterion can balance between exploratory and refinement points in a similar way to the EQI criterion. Recall from the discussion in the last paragraph of Section 3.1 that with $\alpha \geq 0.5$, our EQI criterion generally favors the lowest possible h_{n+1} value. In order to make a sensible choice of h_{n+1}, we need to set $\alpha \geq 0.5$ and bring the cost of computer run at h_{n+1} into the selection criterion. For $\alpha \geq 0.5$, both the EQI part and $C(h)$ increase as h_{n+1} decreases. As $h_{n+1} \to 0$, $C(h)$ tends to infinity while EQI is still bounded. Therefore, the optimum h_{n+1} shall not be very small. On the other hand, as $h_{n+1} \to \infty$, EQI tends to zero but the cost function tends to a positive constant. Therefore, the optimum h_{n+1} shall not be very big, either. In practice, not all h values are realistic and thus we need to confine the choice of h_{n+1} in a certain region. To ease the search for optimum h_{n+1}, we can choose h_{n+1} from a number of prespecified levels.

Let

$$p(\mathbf{x_{n+1}}, h_{n+1}) = [\min Q_n(\mathbf{x_i}) - \{\mathrm{E}(l_{n+1}(\mathbf{x_{n+1}})) + \Phi^{-1}(\alpha) r_{n+1}(\mathbf{x_{n+1}})\}]/\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2}.$$

From (10), the EQI part tends to zero exponentially fast as $p(\mathbf{x_{n+1}}, h_{n+1}) \to -\infty$. Therefore, for refinement runs (i.e., low $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2}$), $r_{n+1}(\mathbf{x_{n+1}})$ has to be very small, which requires small h_{n+1}. On the other hand, for exploratory runs (i.e. high $\mathrm{Var}(l_{n+1}(\mathbf{x_{n+1}}))^{1/2}$),

$r_{n+1}(\mathbf{x_{n+1}})$ can be relatively big. As a result, EQIE generally recommends lower h for refinement runs and higher h for exploratory runs. Similarly, EQIE with a higher (and resp. lower) α tends to choose a lower (and resp. higher) h. Based on a limited simulation study (further discussed in Section 4.1), any choice of α with $0.5 \leq \alpha \leq 0.9$ seems to do equally well.

We now illustrate the EQIE criterion with a toy example. Assume the input space is $[0, 1]$, the choices for h are $\{1, 1.5, 2\}$ and the output function from the simulator is

$$\zeta(x, h) = \left\{ \frac{\sin(20x)}{1+x} + 3x^3 \cos(5x) + 10(x - 0.5)^2 - 0.6 \right\} /2 + h \sin(15\pi(x + 0.1))/5.$$

The objective function (i.e., the exact solution) and the output functions with $h = 1$, $h = 1.5$ and $h = 2$ are depicted in Figure 1. The global minimum for the exact solution is attained at $x = 0.557$.

Suppose we have performed six computer runs with $x = (0.11, 0.28, 0.45, 0.62, 0.79, 0.96)$ and $h = (1, 2, 1, 2, 2, 2)$. The six points for x are equally spaced with an increment of 0.17 and h is chosen randomly from 1 and 2. Figure 2 gives the curves of estimated mean and variance for the objective function and the EQIE curves for $h = 1$, $h = 1.5$ and $h = 2$ with $\alpha = 0.9$.

The three curves are similar except for regions near 0.28, 0.62 and 0.79. The highest EQIE value is obtained at $h = 2$. This is reasonable because low accuracy simulations are more desirable in the beginning of the optimization process and for exploration.

Using the EQIE criterion, we carry out five simulations sequentially with the inputs 0.731, 0.827, 0.336, 0.235 and 0.545. Those inputs are exploratory and thus $h = 2$ is recommended by the EQIE for all five runs. With the eleven runs, the estimated mean and variance for the objective function and the EQIE curves for $h = 1$, $h = 1.5$ and $h = 2$ with $\alpha = 0.9$ are shown in Figure 3. After five new runs, the best combination suggested by the EQIE is $x = 0.544$ and $h = 1.5$. As we proceed further by following the EQIE criterion, the proportion of runs with $h = 1.5$ and $h = 1$ increases.

3.3 Steps For Optimization

In this section, we outline the steps for the optimization process. We propose the following algorithm based on the EQIE criterion:

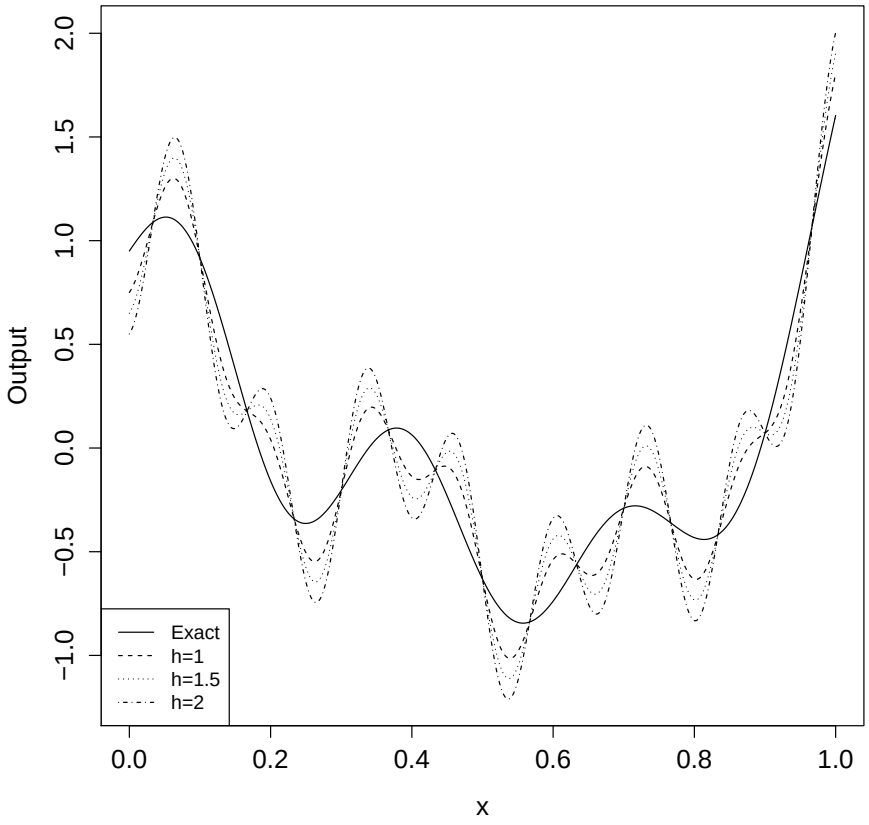

Figure 1: The objective function, $\zeta(\mathbf{x}, 0)$ (solid curve) and the output functions, $\zeta(\mathbf{x}, h)$ (dashed curves), toy example

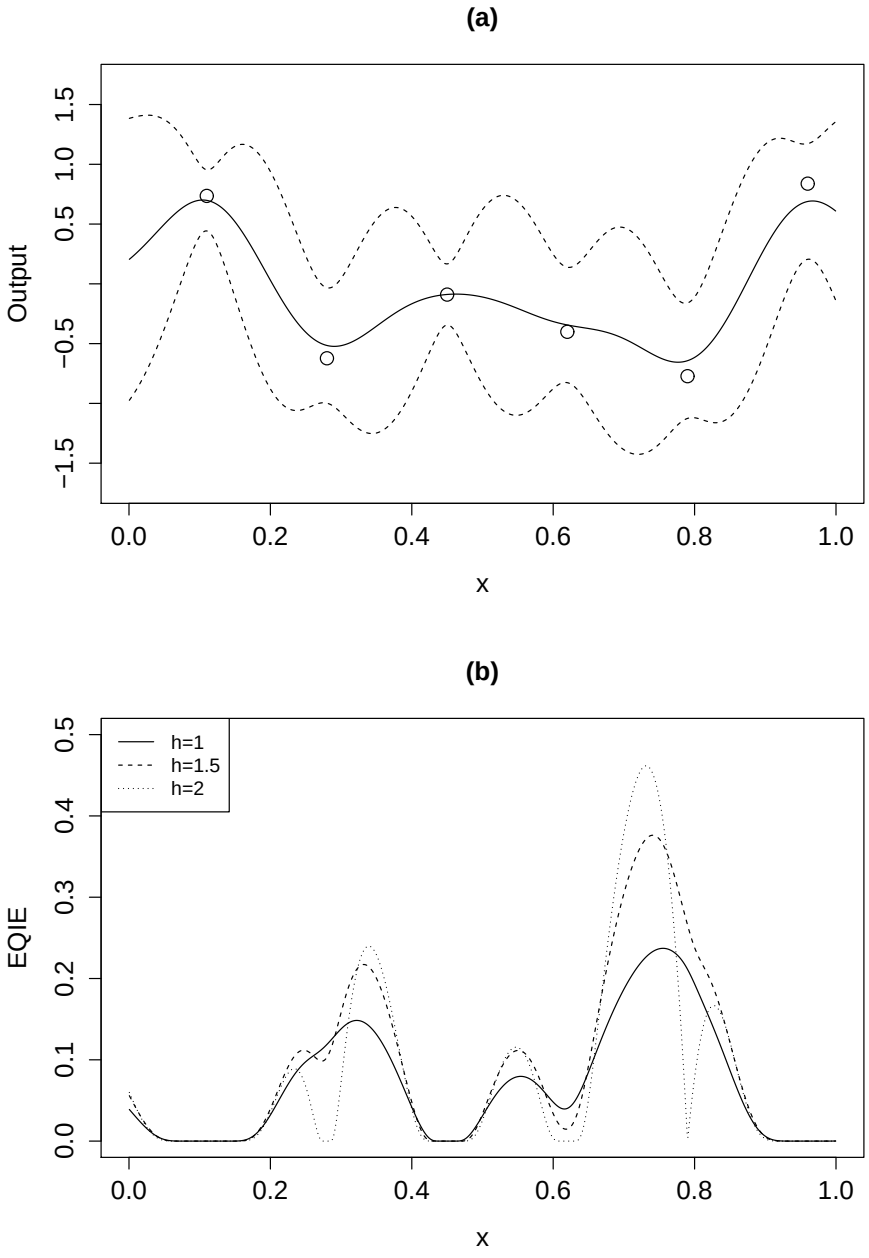

Figure 2: The estimated curves for the objective function and the EQIE curves, toy example. (a) gives the six outputs (circles), the estimated mean objective function (solid curve) and the estimated mean objective function plus/minus 1.96 times of the estimated standard deviation for the mean estimator (dotted curve); (b) gives the EQIE curves for $h = 1$, $h = 1.5$ and $h = 2$.

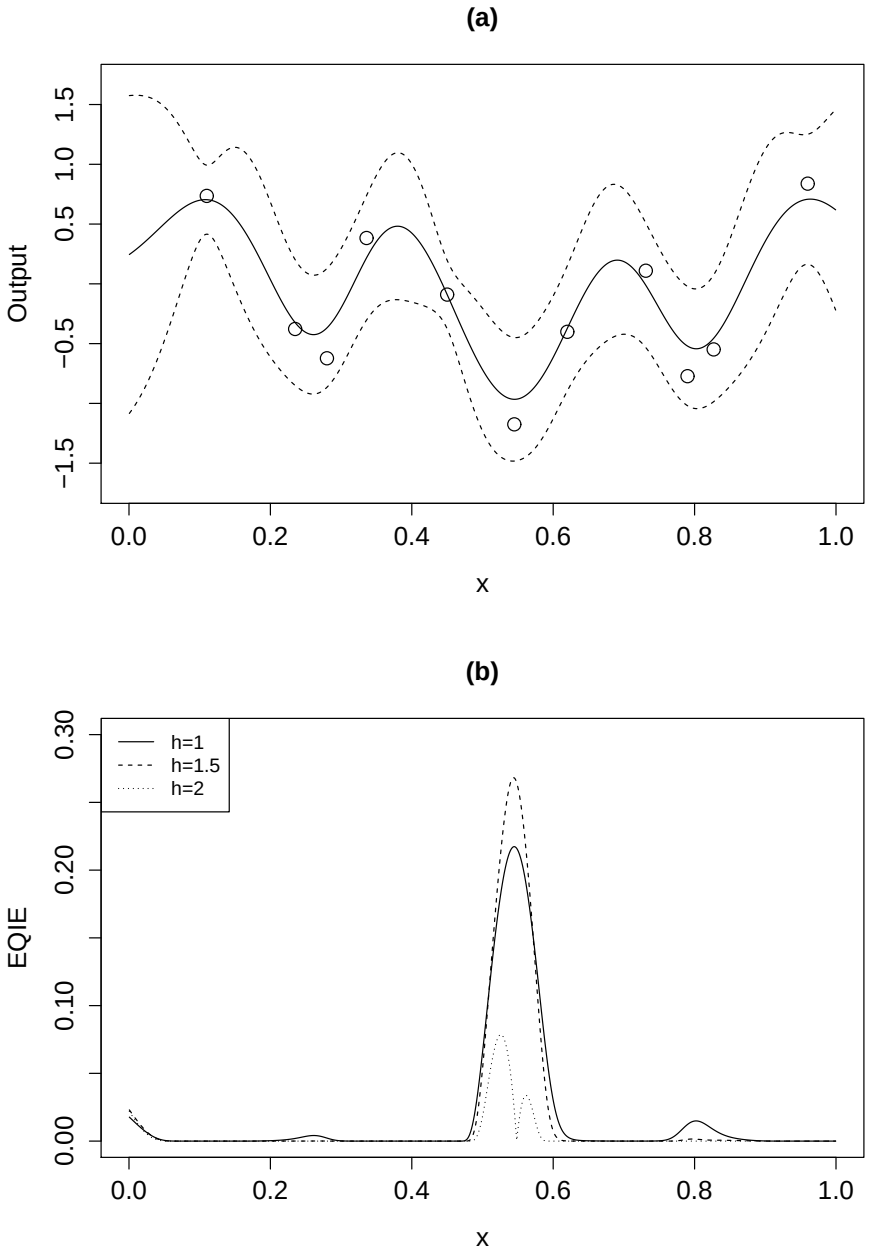

Figure 3: The estimated curves for the objective function and the EQIE curves, toy example. (a) gives the eleven outputs (circles), the estimated mean objective function (solid curve) and the estimated mean objective function plus/minus 1.96 times of estimated standard deviation for the mean estimator (dotted curve); (b) gives the EQIE curves for $h = 1$, $h = 1.5$ and $h = 2$.

1. **Generate initial outputs.** Recall that τ^2/σ^2 gives the ratio for the magnitude of error and the variability of the exact solution and is closely related to the optimum h. Even though we may have prior knowledge on τ^2/σ^2, we want to improve its estimate during the optimization process. For the purpose of fitting τ^2/σ^2, initial runs should be assigned with at least two levels of accuracy. As discussed in the introduction, low accuracy experiments are most suitable for initial runs. Therefore, we recommend the use of two levels of fidelity with the 80%-20% mix for the low and medium accuracy experiments. However, if τ^2/σ^2 is known very well before the simulations, there is no need to include medium accuracy in the initial runs. Since the emulator is fitted jointly by low and medium accuracy runs, it is desirable that all runs come from a space-filling design. We recommend the use of nested space-filling designs (Qian et al., 2009) with the subsample assigned to the medium accuracy.

2. **Fit the Gaussian process model in (3) to data and adopt a cost function $C(h)$.** Tuo et al. (2014) adopt a fully Bayesian approach in which the Markov chain Monte Carlo (MCMC) method is used. In our scheme, we need to refit the model after each run of the computer code and thus a fully Bayesian approach may be too time-consuming. In our experience, a hybrid Bayesian inference by plugging in the posterior mode of the model parameters can be much faster and still has good performance. We use the posterior mode to fit the model with priors for $\boldsymbol{\theta_1}$, $\boldsymbol{\theta_2}$ and τ^2/σ^2 given by the gamma distributions. More discussions on the problem of hyperparameters estimation can be found in Tuo et al. (2014).

3. **Add a new computer run.** Search for the combinations of $(\mathbf{x_{n+1}}, h_{n+1})$ with the highest EQIE value (with $\alpha = 0.9$) and run the computer code with the chosen $(\mathbf{x_{n+1}}, h_{n+1})$.

4. **Re-fit the Gaussian process model.** As in Step 2, we use the posterior mode to fit the model with $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$ and τ^2/σ^2 reestimated.

5. **Iterate between Steps 3 and 4** until the EQIE value goes below a prespecified quantity or the computational budget is used up. We recommend using one third

of the budget for initial simulation runs (in Step 1) and the other two thirds for follow-up runs.

6. **Give the final answer on x.** Compute $Q_N(\mathbf{x_i})$ with $\alpha = 0.9$ for $i = 1, \ldots, N$, where N is the number of computer runs. The $\mathbf{x_i}$ with minimum $Q_N(\mathbf{x_i})$ is the final optimized input. We use $Q_N(\mathbf{x_i})$ to decide the final answer because it considers both the mean estimate on the exact solution and the confidence level of the estimate. (The α value can be chosen in the range $[0.5, 0.9]$.)

It is a folklore in computer experiments that roughly $10d$ points are needed to fit a Gaussian process model with d dimensions (Loeppky et al., 2009). Therefore the number of initial runs may be selected to be close to $10d$. The h value of low accuracy runs can thus be decided. A nested space-filling design is a space-filling design whose subsample is also space-filling. Such designs include nested Latin hypercube designs (Qian, 2009) and nested OA-based Latin hypercube designs (He and Qian, 2011). The cost function can usually be approximated by a simple formula such as $C(h) = bh^c$ or $C(h) = a + bh^c$. The two or three parameters can be fitted by prior computer runs from the same computer program and be updated after new runs. A constant may be added to represent the manual effort associated with each trial. The search of $(\mathbf{x_{n+1}}, h_{n+1})$ in Step 3 is over a finite set. Space-filling designs such as OA-based Latin hypercube designs (Tang, 1993) and some sequential steps can be employed for choosing the finite set. As a result, the added runs may not be the global optimum of the EQIE criterion, especially for high dimensional cases.

The algorithm based on the EQI criterion is the same as the above algorithm except that, in Step 3, we set a relatively high accuracy level for h_{n+1}, search for $\mathbf{x_{n+1}}$ with the highest EQI value ($\alpha = 0.5$) and run the computer code with the chosen $\mathbf{x_{n+1}}$. Computer code written in Matlab that are used for the EQI and EQIE algorithms is given in the supplementary materials.

4 SIMULATION RESULTS

We apply the proposed EQI and EQIE criteria in two simulation studies.

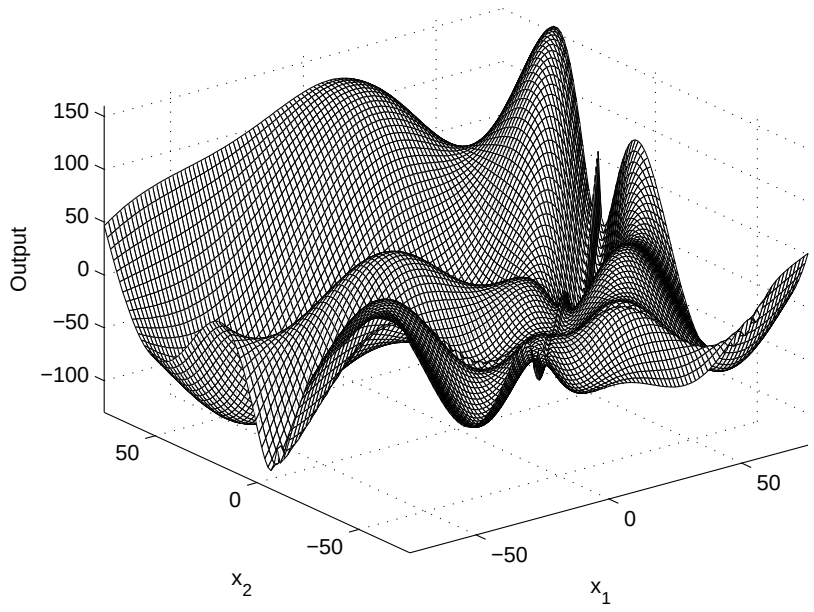

Figure 4: The Eggholder function

4.1 The Eggholder Function

First, we apply the proposed EQI and EQIE criteria to minimize the Eggholder function (Jamil and Yang, 2013):

$$\varphi(\mathbf{x}) = -(x_2 + 47)\sin\left(\sqrt{|x_1/2 + x_2 + 47|}\right) - x_1 \sin\left(\sqrt{|x_1 - x_2 - 47|}\right),$$

where the region for (x_1, x_2) is $[-75, 75] \times [-75, 75]$. A 3D-plot of the function is displayed in Figure 4. The function has many local minima, which makes it hard to optimize.

We assume the error term $\delta(\mathbf{x}, h)$ has the following sine distribution:

$$\delta(\mathbf{x}, h) = 20h \sin(4\pi(x_1/150 + x_2/150 + v)), \tag{13}$$

where v has the uniform distribution on $[0, 1]$. We assume $C(h) = h^{-3}$ and the variance of $\delta(\mathbf{x}, h)$ is linear in h^2, i.e., $w = 2$ in (5). Thus both w and $C(h)$ are known.

For comparison, we try the following five optimization schemes.

EQIE: 25 initial runs and some follow-up runs with varying levels of accuracy. The initial runs come from a nested OA-based Latin hypercube (He and Qian, 2011) of five

levels, in which 5 runs are with $h = 1.2114$ and the other runs are with $h = 1.4057$. The input and the h value of the follow-up runs are chosen by the EQIE criterion ($\alpha = 0.9$), where the choice set for h is $\{1.4057, 1.2114, 1.0000, 0.8500\}$.

EQI: 25 initial runs and 35 follow-up runs. The initial runs are the same as that for the EQIE scheme and the inputs of the follow-up runs are chosen by maximizing the EQI criterion ($\alpha = 0.5$). Since the EQI value cannot be used to recommend h, we manually set $h = 1$ for the follow-up runs.

EI-1: 25 initial runs and 100 follow-up runs by the EI criterion, all with $h = 1.4057$. The initial runs come from an OA-based Latin hypercube (Tang, 1993) of five levels.

EI-2: 16 initial runs and 64 follow-up runs by the EI criterion, all with $h = 1.2114$. The initial runs come from an OA-based Latin hypercube of four levels.

EI-3: 9 initial runs and 36 follow-up runs by the EI criterion, all with $h = 1$. The initial runs come from an OA-based Latin hypercube of three levels.

For each scheme, the total budget is 45, with 9 to 10 units of computation for initial runs and the rest for the follow-up runs. For instance, for EI-1, each run takes $C(h) = (1.4057)^{-3} = 0.36$ unit of time, the 25 initial runs take $0.36 \times 25 = 9$ units of time, and the follow-up runs take $0.36 \times 100 = 36$ units of time. Thus, the total time is $9 + 36 = 45$ units.

We perform 100 simulation trials for each of the five schemes. In each trial, we first sample v in (13) from $U(0, 1)$, then apply the schemes to obtain the optimum inputs. The "response value" refers to the objective function $\varphi(\mathbf{x})$ at the optimum inputs. A plot of the response values as functions of cost, averaged from 100 trials, is given in Figure 5. It is seen that the EQI and EQIE schemes outperform other schemes after 18 units of time, presumably because we use higher h for initial runs and lower h for the follow-up runs. We have also tried the EI scheme whose runs are carried out with $h = 0.7631$, but it was inferior to the EI-3 scheme. Note that the value of 0.7631 was chosen in order to obtain a feasible sample size for OA-based Latin hypercube.

The EQIE scheme is better than the EQI scheme by a small margin after 24 units of time, presumably because the EQIE scheme recommends better h values than simply

setting $h = 1$ for the follow-up runs. It is also seen that the EQIE scheme stabilizes after 30 units of time, roughly three times the cost of the initial runs. This suggests that, for this example, it is optimal to spend one third budget on initial runs.

We have tried three α values (0.5, 0.7, 0.9) for the EQI scheme and four α values (0.5, 0.7, 0.9, 0.95) for the EQIE scheme. The results are given in the supplementary materials in two figures. It appears that both schemes work well in this range of α values and there is no appreciable difference among them. For the consistency of presentation, we use $\alpha = 0.5$ for EQI and $\alpha = 0.9$ for EQIE in the two example in this section and the casting example in Section 5.

The EQIE scheme chooses h values adaptively. Figure 6 summarizes the proportion of runs from the EQIE scheme in each accuracy level, defined by 5 time periods (i.e., by computation time used). This figure illustrates two intuitive features of EQIE. First, since initial runs (i.e., the first period) are exploratory and with high h, the second period should use more refinement runs at a low h. Figure 6 shows that EQIE accomplishes exactly this. Second, as the desired input is approached (roughly at the end of the third period), more exploratory runs (i.e., with higher h) should be used to ensure optimality, which is again observed in Figure 6. Overall, 66% of the follow-up runs are performed at the lowest accuracy ($h = 1.4057$), which require 40% computation time, and 12% of the follow-up runs are performed at the highest accuracy ($h = 0.85$), which require 33% computation time. The remaining runs take the two middle values $h = 1.2114$ and 1.

4.2 The 4-dimensional Ackley Function

Next, we apply the proposed EQI and EQIE criteria to minimize the 4-dimensional Ackley function (Ackley, 1987):

$$\varphi(\mathbf{x}) = -20 \exp\left(-0.2\sqrt{\frac{1}{4}\sum_{i=1}^{4} x_i^2}\right) - \exp\left(\frac{1}{4}\sum_{i=1}^{d}\cos(2\pi x_i)\right) + 20 + \exp(1),$$

where the region for (x_1, x_2, x_3, x_4) is $[0, 1]^4$. A 3D-plot of the function when holding $x_3 = 0$ and $x_4 = 0$ is displayed in Figure 7.

We assume the error term $\delta(\mathbf{x}, h)$ has the following sine distribution:

$$\delta(\mathbf{x}, h) = 0.2h \sin(4\pi(x_1 + x_2 + x_3 + x_4 + v)), \tag{14}$$

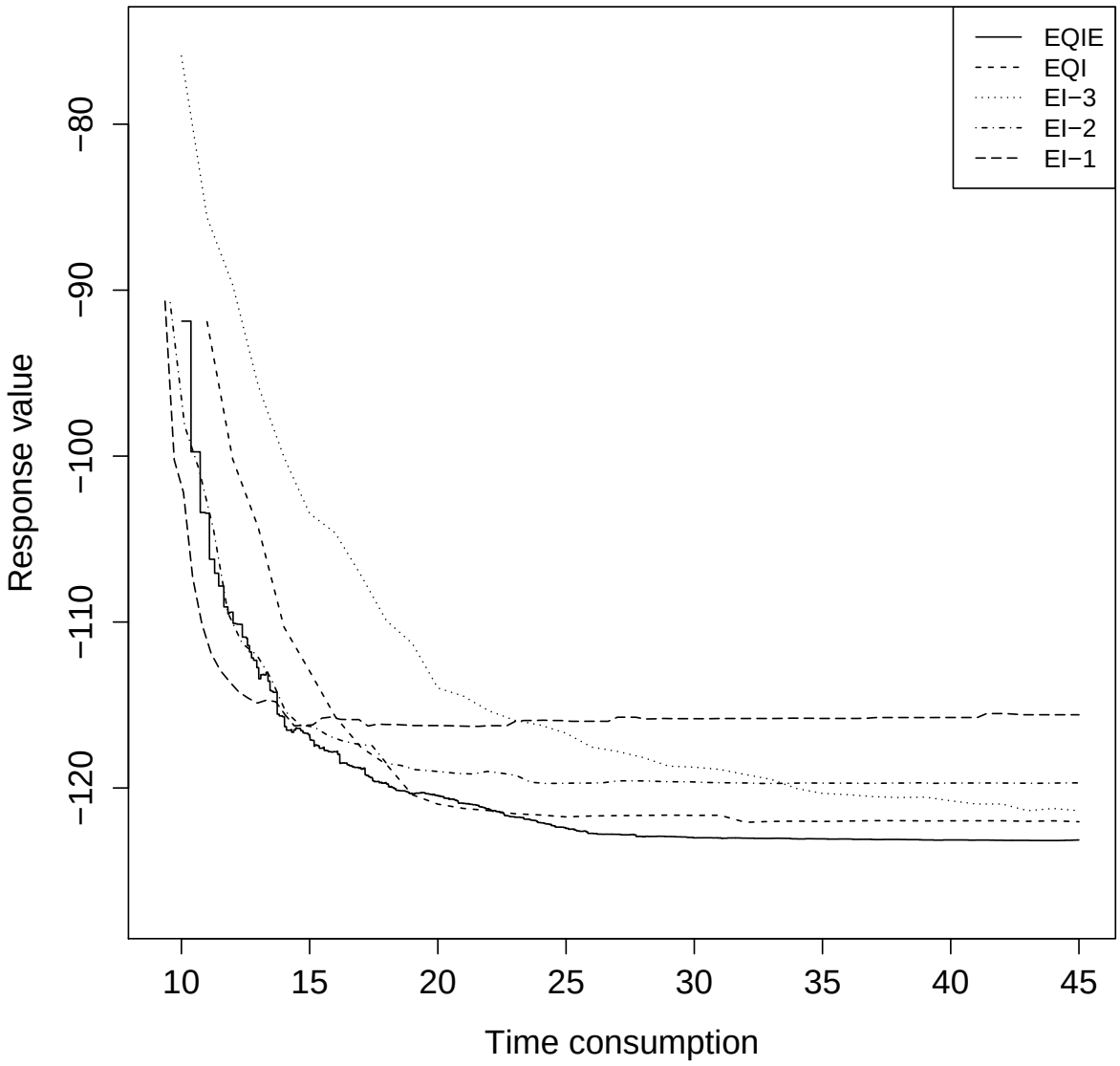

Figure 5: The response values as functions of cost, Eggholder function

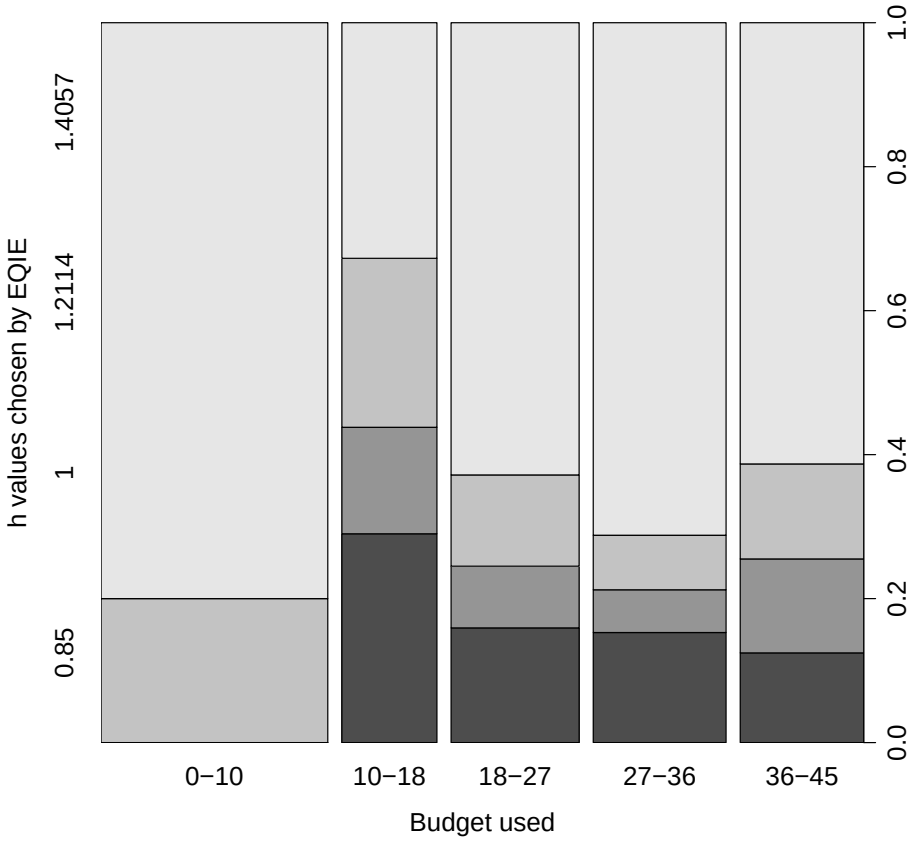

Figure 6: The proportion of runs in each accuracy level, defined by 5 time periods, Egghold-er function. The color shade from light to dark means high h to low h.

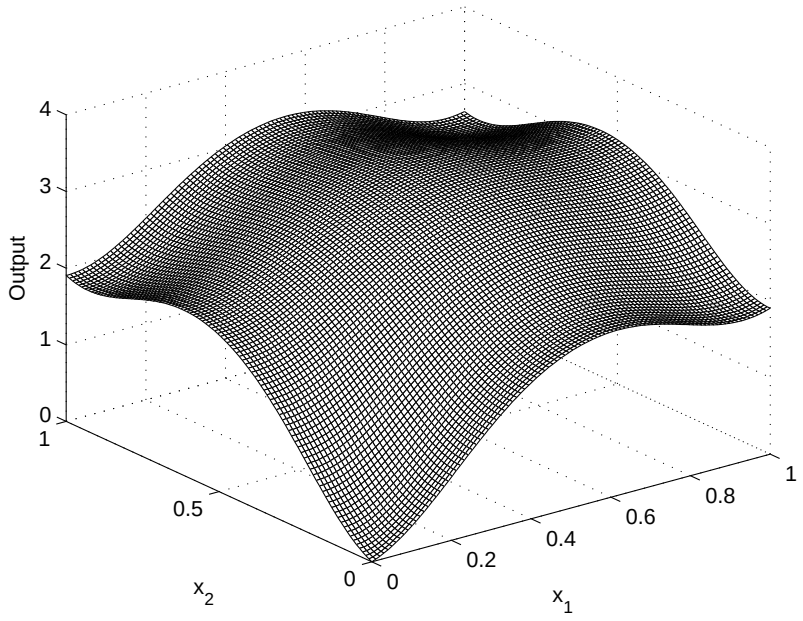

Figure 7: The 4-dimensional Ackley function when $x_3 = 0$ and $x_4 = 0$

where v has the uniform distribution on $[0, 1]$. We assume $C(h) = h^{-3}$ and the variance of $\delta(\mathbf{x}, h)$ is linear in h^2, i.e., $w = 2$ in (5). Thus both w and $C(h)$ are known.

As in Section 4.1, we consider the same five schemes using EQIE ($\alpha = 0.9$), EQI ($\alpha = 0.5$) and EI. The tuning parameter for EI-1, EI-2 and EI-3 are $h = 1.3572$, $h = 1.1817$ and $h = 1.0$, respectively, and the initial inputs are generated from a Latin hypercube of 50, 33 and 20 levels, respectively. For EQI and EQIE, the initial runs are generated from a nested Latin hypercube (Qian, 2009) of 50 levels, 10 runs of which with $h = 1.1817$ and the rests with $h = 1.3572$. For the follow-up runs, the choice set of h for EQIE is $\{1.3572, 1.1817, 1.0000, 0.8500\}$ and h is set to 1 for EQI. The total budget is 100 for all schemes.

We perform 100 simulation trials for each of the five schemes. In each trial, we first sample v in (14) from $U(0, 1)$, then apply the schemes to obtain the optimum inputs. A plot of the response values as functions of cost, averaged from 100 trials, is given in Figure 8. Again, EQIE and EQI perform significantly better than EI with any accuracy, showing the advantage of using multi-fidelity computer runs. The EQIE scheme is better than the EQI

scheme before 64 units of time, but both schemes find near optimal inputs (with response value less than 0.1) after 66 units of time. We have also tried the EI scheme whose runs are carried out with $h = 0.8434$. It outperformed the EI-3 scheme after 85 units of time but is uniformly inferior to the EQI and EQIE schemes. Figure 9 reports the proportion of runs from the EQIE scheme in 5 periods, arranged in a similar way to Figure 6. Again, many refinement runs at low h are used in the second period to compensate the initial runs with high h, and more exploratory runs are used after the desired input is reached roughly at the end of the third period. These features suggest the EQIE scheme chooses reasonable levels of h.

5 A CASTING EXAMPLE

In this section, we illustrate the proposed method with an alloy casting problem.

Alloy produced by casting operations are prone to shrinkage defects if the manufacturing process is not properly designed. Computer programs are commonly used to simulate casting processes. The optimal engineering design reached from computer experiments is usually used as the first design for physical experiments. In this example, we use a commercial software called Magma (website: http://www.magmasoft.com/en/) to study the casting operation of an aluminum alloy part. For Magma, the physical model is based on the coupling of the thermal and fluent dynamics in terms of a set of partial differential equations, which is solved numerically by finite element methods. One such computer experiment may take minutes to days to finish, depending on the mesh size. Finer meshes are required for more sophisticated castings. Although our methods are more applicable for complex castings for which one trial takes at least several hours, for illustrative purpose, we consider the optimization of a moderately sophisticated casting example where one trial can be completed within ten minutes. Refer to Stefanescu (2008) for a detailed discussion.

Among numerous outputs produced by Magma, the Niyama criterion (Niyama et al., 1982) is frequently used to represent the vulnerability to shrinkage defects. For each physical position, the Niyama criterion is defined by $Ni = G/\sqrt{R}$, where G is the temperature gradient and R is the cooling rate, both evaluated at the time when the liquid solidifies. Regions with $Ni < 0.3$ indicate a high risk for shrinkage defects. Our goal is to minimize

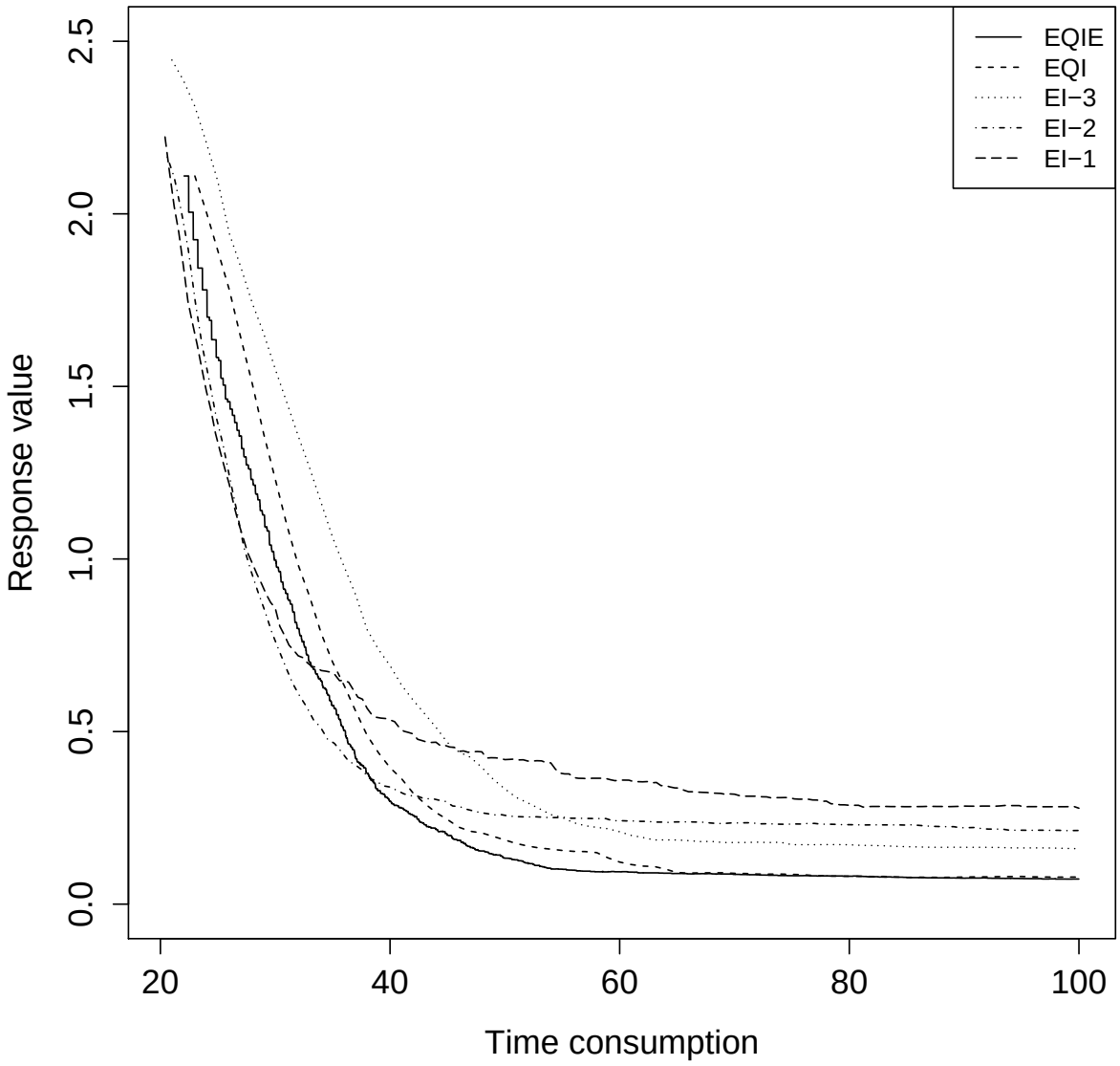

Figure 8: The response values as functions of cost, Ackley function.

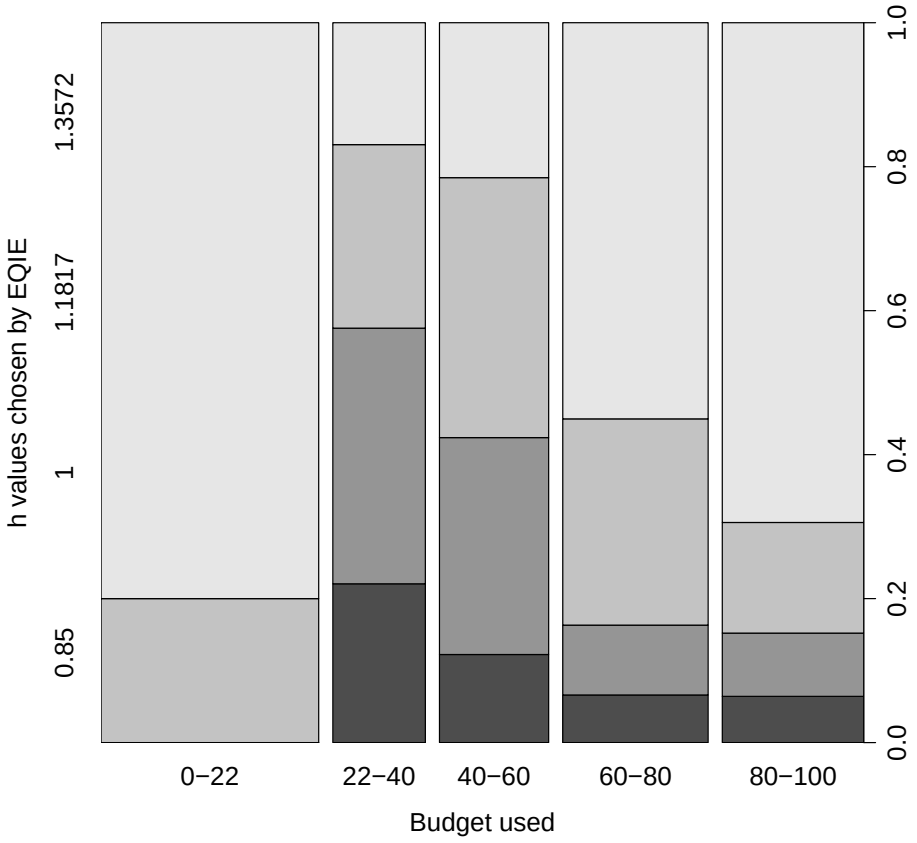

Figure 9: The proportion of runs in each accuracy level, defined by 5 time periods, Ackley function. The color shade from light to dark means high h to low h.

y, the percentage of volume with $Ni < 0.3$. To prevent shrinkage, it is common to place several feeders in the mold. A feeder is a reservoir built on the top of the cast. As castings shrink upon cooling, liquid contained in the feeders can flow back to fill the gap. In this example, we place four feeders on the top of the cast. The four feeders combined need to cover as much volume of the cast as possible. Therefore, the position of feeders is critical to the degree of shrinkage defects. We set one feeder fixed and use three control variables to represent the sites of the other feeders. The tuning parameter h is the length of mesh cubes which can be 6.49, 5.79, 5.11 or 4.57 millimeters. From our experience with Magma, one run takes approximately $20000h^{-5}$ minutes. Since Magma uses a finite element method, we use $w = 2$ based on the discussion in Section 2.2. Informative priors are used for the model parameters.

We consider five schemes using EI, EQI ($\alpha = 0.5$) or EQIE ($\alpha = 0.9$). For EI-1, EI-2 and EI-3, 69, 39 and 21 runs are carried out with $h = 6.49$, $h = 5.79$ and $h = 5.11$, respectively. Their initial designs are Latin hypercube designs with 23, 13 and 7 levels, respectively. The initial design for EQIE and EQI consists of 16 runs with $h = 6.49$ and 4 runs with $h = 5.79$, generated from a nested Latin hypercube design. EQI has 14 follow-up runs with $h = 5.11$ and for EQIE, the levels of h are chosen from $\{6.49, 5.79, 5.11, 4.57\}$ by maximizing the EQIE criterion. The budget is the same for all schemes with one third computation time spent on initial runs.

We provide the values of x, h and y for the runs from the EQI and EQIE schemes in Tables 1 and 2. The data sets for all five methods are given in supplementary files. Table 3 reports the optimal inputs returned by the five schemes and their corresponding output values evaluated at a high accuracy level ($h = 3.98$). From the results, EQI is slightly better than EQIE, and both provide much better performance than EI methods. Levels with $h = 6.48$, $h = 5.79$ and $h = 4.57$ are chosen from EQIE for 25, 2 and 3 times, respectively, while the level $h = 5.11$ is never chosen. The 26th, 29th and 30th runs are set with $h = 4.57$, presumably because they are refinement runs. Indeed, they are close to the 24th and 25th runs. The final returned x is the site for the 31th run, which locates in between of the 29th and 30th runs. Presumably because this site has already been much investigated by runs 24-26 and 29-31, most of the later runs are exploratory. From the

Table 1: Data points for the EQI scheme, casting example. The first 20 runs are initial runs.

	x_1	x_2	x_3	h	y		x_1	x_2	x_3	h	y
1	0.175	0.289	0.373	6.49	5.10	18	0.353	0.151	0.412	5.79	5.74
2	0.444	0.892	0.076	6.49	2.77	19	0.700	0.815	0.937	5.79	4.43
3	0.278	0.218	0.984	6.49	8.06	20	0.992	0.712	0.693	5.79	4.86
4	0.865	0.126	0.792	6.49	6.21	21	0.524	0.593	0.146	5.11	1.58
5	0.216	0.076	0.199	6.49	6.76	22	0.622	0.498	0.138	5.11	2.79
6	0.489	0.013	0.573	6.49	5.62	23	0.428	0.615	0.159	5.11	1.70
7	0.599	0.405	0.836	6.49	3.89	24	0.460	0.609	0.057	5.11	1.96
8	0.301	0.974	0.621	6.49	3.87	25	0.473	0.614	0.200	5.11	1.41
9	0.934	0.795	0.281	6.49	4.66	26	0.480	0.618	0.218	5.11	1.61
10	0.626	0.923	0.860	6.49	4.92	27	0.489	0.472	0.151	5.11	1.28
11	0.045	0.558	0.732	6.49	5.41	28	0.529	0.418	0.013	5.11	1.63
12	0.831	0.346	0.348	6.49	3.01	29	0.513	0.459	0.077	5.11	1.28
13	0.661	0.624	0.515	6.49	4.06	30	0.636	0.276	0.104	5.11	1.74
14	0.078	0.354	0.047	6.49	5.58	31	0.780	0.236	0.046	5.11	2.46
15	0.136	0.548	0.247	6.49	3.99	32	0.566	0.382	0.092	5.11	1.56
16	0.775	0.680	0.479	6.49	4.49	33	0.556	0.361	0.101	5.11	1.69
17	0.502	0.492	0.146	5.79	1.49	34	0.610	0.389	0.052	5.11	1.57

results, some of the refinement runs are assigned with low h ($h = 4.57$) and all exploratory runs are assigned with high h.

6 CONCLUSIONS AND FURTHER DISCUSSION

This paper addresses the problem of kriging-based optimization for deterministic computer experiments with multiple levels of fidelity (or accuracy). The key of the optimization algorithm is to choose a criterion for scoring candidate inputs and accuracy level. We propose the EQIE criterion that scores candidate combinations of inputs and accuracy.

Table 2: Data points for the EQIE scheme, casting example. The first 20 runs are initial runs. For the remaining 30 runs, levels for h are chosen by the EQIE criterion.

	x_1	x_2	x_3	h	y		x_1	x_2	x_3	h	y
1	0.175	0.289	0.373	6.49	5.10	26	0.519	0.525	0.199	4.57	1.54
2	0.444	0.892	0.076	6.49	2.77	27	0.212	0.992	0.256	6.49	5.23
3	0.278	0.218	0.984	6.49	8.06	28	0.639	0.849	0.000	6.49	3.85
4	0.865	0.126	0.792	6.49	6.21	29	0.510	0.463	0.225	4.57	1.37
5	0.216	0.076	0.199	6.49	6.76	30	0.560	0.410	0.315	4.57	1.63
6	0.489	0.013	0.573	6.49	5.62	31	0.526	0.468	0.242	6.49	2.20
7	0.599	0.405	0.836	6.49	3.89	32	0.467	0.480	0.349	6.49	2.12
8	0.301	0.974	0.621	6.49	3.87	33	0.997	0.027	0.167	6.49	5.61
9	0.934	0.795	0.281	6.49	4.66	34	0.580	0.998	0.292	6.49	5.03
10	0.626	0.923	0.860	6.49	4.92	35	0.107	1.000	0.976	6.49	6.18
11	0.045	0.558	0.732	6.49	5.41	36	0.514	0.483	0.272	6.49	1.84
12	0.831	0.346	0.348	6.49	3.01	37	0.602	0.512	0.002	5.79	3.23
13	0.661	0.624	0.515	6.49	4.06	38	1.000	0.448	0.045	6.49	4.28
14	0.078	0.354	0.047	6.49	5.58	39	0.715	0.384	0.561	5.79	3.09
15	0.136	0.548	0.247	6.49	3.99	40	0.868	0.488	0.999	6.49	7.20
16	0.775	0.680	0.479	6.49	4.49	41	0.414	0.669	0.998	6.49	5.13
17	0.502	0.492	0.146	5.79	1.49	42	0.001	0.941	0.570	6.49	6.67
18	0.353	0.151	0.412	5.79	5.74	43	0.528	0.444	0.245	6.49	2.04
19	0.700	0.815	0.937	5.79	4.43	44	0.542	0.304	0.198	6.49	1.89
20	0.992	0.712	0.693	5.79	4.86	45	0.000	0.926	0.003	6.49	6.69
21	0.512	0.611	0.147	6.49	2.06	46	0.994	0.990	0.000	6.49	7.12
22	0.626	0.433	0.145	6.49	2.96	47	0.514	0.412	0.249	6.49	2.29
23	0.413	0.530	0.143	6.49	2.59	48	0.658	0.185	0.004	6.49	3.76
24	0.514	0.508	0.269	6.49	1.83	49	0.000	0.101	0.536	6.49	8.25
25	0.513	0.517	0.189	6.49	1.72	50	0.714	0.162	0.303	6.49	4.33

Table 3: Optimal inputs recommended by the five schemes and the associated output values obtained with $h = 3.98$

Method	Input	Output
EI-1	(0.582, 0.443, 0.347)	1.566
EI-2	(0.516, 0.488, 0.347)	1.448
EI-3	(0.392, 0.809, 0.776)	1.524
EQI	(0.513, 0.459, 0.077)	1.321
EQIE	(0.526, 0.468, 0.242)	1.352

Based on the EQIE criterion, we propose an optimization scheme that sequentially adds new computer runs.

Our criterion can be seen as an extension of the well-known expected improvement (EI) criterion to multi-fidelity experiments. The EQIE criterion consists of two parts, the EQI criterion and a cost function. The EQI criterion is an adaption of the EQI criterion proposed by Picheny et al. (2013) to the model by Tuo et al. (2014). Given reasonable levels of mesh size, the input that maximizes the EQI criterion is a good candidate for follow-up runs. However, the EQI criterion itself cannot recommend the levels of mesh size for follow-up runs. By incorporating a cost function, the EQIE criterion can recommend both input value and level of mesh size for follow-up runs.

The EQIE scheme employs experiments with three or more levels of accuracy, with relatively low fidelity used for initial runs. From the simulation results, a reasonable strategy for sample size allocation is to use one third of the computation resource on the initial runs and the rest for the added runs. The process shall stop when the EQIE value goes below a prespecified quantity or the computational budget is used up. Simulation results and a real example show that the EQIE scheme outperforms schemes using EI. Thus, we conclude that it is advantageous to employ multi-fidelity experiments in optimization and to assign different levels of mesh size for follow-up experiments. In particular, later experiments should in general have higher level of accuracy than earlier experiments, and refinement experiments should in general have higher level of accuracy than exploratory experiments.

In our work, the errors are assumed to be realizations of another deterministic Gaussian

process. There are other models for emulating multi-fidelity deterministic computer experiments, such as Kennedy and O'Hagan (2000), Reese et al. (2004), Qian and Wu (2008). These models treat the tuning parameter as a qualitative factor. On the other hand, the TWY model treats the tuning parameter as a quantitative variable. As a result, for simulators with more than two levels of accuracy, this latter model requires fewer parameters to be fit than other methods. Unlike the other models, the TWY model can be used to predict for outcomes with untried fidelity. Since our method uses computer runs with increasing levels of accuracy, the TWY model is more suitable. On the other hand, if the computer code allows only two accuracy levels, or if the variances of all feasible accuracy levels are known before the experiment, other models can be considered.

Although the EQI and EQIE criteria allow the tuning parameter to be chosen from a continuum of fidelities, we use three or four accuracy levels in our examples. From our limited experience, there is little gain from using more levels. Extension of the current approach to handle continuous scale is beyond the scope of the paper. Finally, regarding the choice of the α value in the EQI and EQIE schemes, there is generally no optimal choice as they involve complicated functions and multiple objectives. The same is true for the original EQI criterion proposed by Picheny et al. (2013). Our limited simulation study seems to suggest that there is a broad range of values between 0.5 and 0.9 that can be chosen by the users. But the user may decide to make his/her choice based on additional information of the specific experiment. Another future research problem is to extend the current method for the purpose of prediction.

7 ACKNOWLEDGMENTS

The authors are grateful to the referees, associate editor, Yuanzhen He, Shifeng Xiong, V. Roshan Joseph and Simon Mak for their valuable comments. Wu's work is supported by NSF DMS-1308424 and DOE DE-SC0010548. Tuo's work is also supported by the National Center for Mathematics and Interdisciplinary Sciences, CAS and NSFC 11271355. He's work is also supported by NSFC 11501550.

Supplementary Materials

Derivations: Our derivations for EQI($\mathbf{x_{n+1}}, h_{n+1}$) in (7) and (10) and EQIE($\mathbf{x_{n+1}}, h_{n+1}$) in (12). (pdf)

Computer code: Computer code of the EQI and EQIE methods written in Matlab for our simulation example in Section 4.1. (GNU zipped tar file)

Data: Data for the casting example in Section 5. (GNU zipped tar file)

Figures for α: Simulation results on $\alpha = 0.5$, $\alpha = 0.7$ and $\alpha = 0.9$ for the EQI scheme and $\alpha = 0.5$, $\alpha = 0.7$, $\alpha = 0.9$ and $\alpha = 0.95$ for the EQIE scheme. (pdf)

References

Ackley, D. H. (1987). *A Connectionist Machine for Genetic Hillclimbing*. Kluwer Boston Inc., Hingham, MA.

Brenner, S. C. and L. R. Scott (2007). *The Mathematical Theory of Finite Element Methods (3rd ed.)*. New York: Springer.

Chkrebtii, O. A., D. A. Campbell, M. A. Girolami, and B. Calderhead (2014). Bayesian uncertainty quantification for differential equations. arXiv:1306.2365.

Forrester, A., A. Sóbester, and A. Keane (2007). Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science 463*, 3251–3269.

He, X. and P. Z. G. Qian (2011). Nested orthogonal array-based Latin hypercube designs. *Biometrika 98*(3), 721–731.

Huang, D., T. Allen, W. Notz, and N. Zeng (2006). Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of Global Optimization 34*, 441–466.

Jamil, M. and X.-S. Yang (2013). A literature survey of benchmark functions for global optimization problems. *Int. Journal of Mathematical Modelling and Numerical Optimisation 4*(2), 150–194.

Jones, D. R., M. Schonlau, and W. J. Welch (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization 13*(4), 455–492.

Kennedy, M. C. and A. O'Hagan (2000). Predicting the output from a complex computer code when fast approximation are available. *Biometrika 87*, 1–13.

Loeppky, J. L., J. Sacks, and W. J. Welch (2009). Choosing the sample size of a computer experiment: A practical guide. *Technometrics 51*(4), 366–376.

Mckay, M. D., W. J. Conover, and R. J. Beckman (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics 21*, 239–245.

Niyama, E., T. Uchida, M. Morikawa, and S. Saito (1982). A method of shrinkage prediction and its application to steel casting practice. *International Cast Metals Journal 7*, 52634.

Picheny, V. and D. Ginsbourger (2013). A nonstationary space-time gaussian process model for partially converged simulations. *SIAM/ASA J. Uncertainty Quantification 1*, 57–78.

Picheny, V., D. Ginsbourger, Y. Richet, and G. Caplin (2013). Quantile-based optimization of noisy computer experiments with tunable precision (with discussions). *Technometrics 55*(1), 2–13.

Qian, P. Z. G. (2009). Nested Latin hypercube designs. *Biometrika 96*(4), 957–970.

Qian, P. Z. G., B. Tang, and C. F. J. Wu (2009). Nested space-filling designs for computer experiments with two levels of accuracy. *Statistica Sinica 19*, 287–300.

Qian, P. Z. G. and C. F. J. Wu (2008). Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments. *Technometrics 50*, 192–204.

Reese, S., A. Wilson, M. Hamada, H. Martz, and K. Ryan (2004). Integrated analysis of computer and physical experiments. *Technometrics 46*, 153–164.

Sacks, J., W. J. Welch, T. J. Mitchell, and H. P. Wynn (1989). Design and analysis of computer experiments. *Statist. Sci. 4*, 409–435.

Santner, T. J., B. J. Williams, and W. I. Notz (2003). *The Design and Analysis of Computer Experiments*. New York: Springer.

Stefanescu, D. M. (2008). *Science and Engineering of Casting Solidification (2nd ed.)*. New York: Springer.

Tang, B. (1993). Orthogonal array-based Latin hypercubes. *Journal of the American Statistical Association 88*(424), 1392–1397.

Tuo, R., C. F. J. Wu, and D. Yu (2014). Surrogate modeling of computer experiments with different mesh densities. *Technometrics 56*(3), 372–380.